John von Neumann Institute for Computing

Parallel Newton-Type Iterative Methods Based on ILU Factorizations

J. Arnal, H. Migallón, V. Migallón, J. Penadés

published in

Parallel Computing:
Current & Future Issues of High-End Computing,
Proceedings of the International Conference ParCo 2005,
G.R. Joubert, W.E. Nagel, F.J. Peters, O. Plata, P. Tirado, E. Zapata (Editors),
John von Neumann Institute for Computing, Jülich,
NIC Series, Vol. **33**, ISBN 3-00-017352-8, pp. 245-252, 2006.

http://www.fz-juelich.de/nic-series/volume33

Parallel Newton-type iterative methods based on ILU factorizations

Josep Arnal[a], Héctor Migallón[b], Violeta Migallón[a], José Penadés[a]

[a]Departamento de Ciencia de la Computación e Inteligencia Artificial,
Universidad de Alicante, E-03071 Alicante, Spain, {arnal, violeta, jpenades}@dccia.ua.es

[b]Departamento de Física y Arquitectura de Computadores,
Universidad Miguel Hernández, 03202 Elche, Alicante, Spain, hmigallon@umh.es

Parallel iterative algorithms based on the Newton method and on two of its variations, the Shamanskii method and the Chord method, for solving nonlinear systems are proposed. These algorithms are based on the two–stage multisplitting methods. Concretely, in order to construct the inner splittings, incomplete LU factorizations are considered. Convergence properties of these parallel methods are analyzed and computational results on two parallel computing systems are discussed. In order to illustrate the behaviour of the proposed algorithms, we have considered a nonlinear elliptic partial differential equation, known as the Bratu problem, which comes from a simplification of the solid fuel ignition model in thermal combustion theory. The reported experiments show the effectiveness of these methods.

1. Introduction

Consider the problem of solving a nonlinear system of the form $F(x) = 0$, where $F : \mathbb{R}^n \to \mathbb{R}^n$ is a nonlinear mapping. Considering that there exists a solution x^* of this system, we can use for solving it the classical Newton method (cf. [10], [7]). Given an initial vector $x^{(0)}$, this method produces the following sequence of vectors

$$x^{(\ell+1)} = x^{(\ell)} - \delta_x^{(\ell)}, \quad \ell = 0, 1, \ldots, \tag{1}$$

where $\delta_x^{(\ell)}$ is the solution of the linear system

$$F'(x^{(\ell)})z = F(x^{(\ell)}), \tag{2}$$

and $F'(x)$ denotes the Jacobian matrix.

Iterative methods can be used for the solution of (2). In this case we are in the presence of a Newton iterative method. Descriptions of these methods can be found, e.g., in [10]. In order to generate efficient algorithms to solve the nonlinear system $F(x) = 0$ on a parallel computer, in [1] it was constructed a parallel Newton iterative algorithm, in which the approximations of the linear systems (2) are accomplished by using a two–stage multisplitting method [5].

In order to construct these methods, let us consider for each x, a two–stage multisplitting of $F'(x)$, $\{P_k(x), Q_k(x), M_k(x), N_k(x), E_k\}_{k=1}^p$, that is, a collection of matrices such that $F'(x) = P_k(x) - Q_k(x),\ 1 \le k \le p$, are splittings of $F'(x)$, called outer splittings, $P_k(x) = M_k(x) - N_k(x),\ 1 \le k \le p$, are splittings of $P_k(x)$, called inner splittings, and $E_k,\ 1 \le k \le p$, are diagonal nonnegative weighting matrices such that $\sum_{k=1}^p E_k = I$.

Let us further consider two sequences of integers. The sequence $m_\ell,\ \ell = 0, 1, \ldots$, indicates the number of linear steps performed to approximate the linear system (2) at the global nonlinear iteration ℓ, and the sequence of non–stationary parameters $q(\ell, s, k),\ \ell = 0, 1, \ldots,\ s = 1, 2, \ldots, m_\ell,\ 1 \le k \le p$, indicates the number of inner steps which processor k carries out at the s outer linear step and at the global iteration ℓ.

Given an initial vector $x^{(0)}$, that satisfies some initial conditions that we will analyze later on, we construct a sequence of vectors $\{x^{(\ell)}\}_{\ell=0}^{\infty}$ in the following way. In order to approximate the linear system $F'(x^{(\ell)})z = F(x^{(\ell)}),\ \ell = 0, 1, \ldots$, setting $z^{(0)} = 0$, we compute the iterates $z^{(s)} = \sum_{k=1}^{p} E_k z_k^{(q(\ell,s,k))},\ s = 1, 2, \ldots, m_\ell$, where $z_k^{(q(\ell,s,k))}$ is an approximation to the solution of the linear system $P_k(x^{(\ell)})z_k = Q_k(x^{(\ell)})z^{(s-1)} + F(x^{(\ell)})$, obtained by computing $q(\ell, s, k)$ iterations of the iterative method based on the splitting $P_k(x^{(\ell)}) = M_k(x^{(\ell)}) - N_k(x^{(\ell)})$, taking as initial vector $z_k^{(0)} = z^{(s-1)}$. That is, $z_k^{(q(\ell,s,k))}$ is achieved by performing the following $q(\ell, s, k)$ iterations, $z_k^{(m+1)} = M_k(x^{(\ell)})^{-1}N_k(x^{(\ell)})z_k^{(m)} + M_k(x^{(\ell)})^{-1}\left(Q_k(x^{(\ell)})z^{(s-1)} + F(x^{(\ell)})\right),\ m = 0, 1, \ldots, q(\ell, s, k) - 1$.

Finally, the next global iterate is computed as

$$x^{(\ell+1)} = x^{(\ell)} - \delta_x^{(\ell)},\ \ell = 0, 1, \ldots, \text{ where } \delta_x^{(\ell)} = z^{(m_\ell)}. \tag{3}$$

In order to study the convergence of this method, let us denote, for $\ell = 0, 1, \ldots,\ s = 1, 2, \ldots, m_\ell$, $1 \le k \le p$,

$$T_k^{\ell,s}(x) = \left(M_k(x)^{-1}N_k(x)\right)^{q(\ell,s,k)} + \sum_{i=0}^{q(\ell,s,k)-1} \left(M_k(x)^{-1}N_k(x)\right)^i M_k(x)^{-1}Q_k(x) \tag{4}$$

$$H_{\ell,s}(x) = \sum_{k=1}^{p} E_k T_k^{\ell,s}(x),\ \ B_{\ell,s}(x) = \sum_{k=1}^{p} E_k \sum_{i=0}^{q(\ell,s,k)-1} \left(M_k(x)^{-1}N_k(x)\right)^i M_k(x)^{-1} \tag{5}$$

Thus $z^{(s)} = H_{\ell,s}(x^{(\ell)})z^{(s-1)} + B_{\ell,s}(x^{(\ell)})F(x^{(\ell)}),\ s = 1, 2, \ldots, m_\ell$, with $z^{(0)} = 0$. From this expression it is easy to deduce $z^{(s)} = \left(\sum_{i=1}^{s-1} \prod_{j=i+1}^{s} H_{\ell,j}(x^{(\ell)})B_{\ell,i}(x^{(\ell)}) + B_{\ell,s}(x^{(\ell)})\right) F(x^{(\ell)})$, where $\prod_{j=i+1}^{s} H_{\ell,j}(x)$ denotes the product of the matrices $H_{\ell,j}(x)$, in the order $H_{\ell,s}(x) \cdots H_{\ell,i+1}(x)$. With this notation the parallel Newton two–stage iterative method (3) can be written as follows

$$x^{(\ell+1)} = G_{\ell,m_\ell}(x^{(\ell)}),\ \ell = 0, 1, \ldots, \tag{6}$$

where $G_{\ell,m_\ell}(x) = x - A_{\ell,m_\ell}(x)F(x)$, and

$$A_{\ell,m_\ell}(x) = \sum_{i=1}^{m_\ell-1} \prod_{j=i+1}^{m_\ell} H_{\ell,j}(x)B_{\ell,i}(x) + B_{\ell,m_\ell}(x). \tag{7}$$

The experiments displayed in [1] show the good behaviour of these methods. In Section 2 we analyze the convergence of these methods when incomplete LU (ILU) factorizations are used in order to obtain the inner splittings. Moreover, in order to reduce the computational cost of each non–linear iteration, we describe and analyze two acceleration techniques for this parallel Newton iterative method, based on both the Chord method and the Shamanskii method. In Section 3 we present some numerical experiments, which illustrate the performance of these algorithms. Previously, in the rest of this section we present some notation, definitions and preliminary results to which we refer later.

A matrix A is said to be a nonsingular M–matrix if A has all nonpositive off–diagonal entries and it is monotone, i.e., $A^{-1} \ge O$. For any matrix $A = (a_{ij}) \in \mathbb{R}^{n\times n}$, we define its comparison matrix $\langle A\rangle = (\alpha_{ij})$ by $\alpha_{ii} = |a_{ii}|,\ \alpha_{ij} = -|a_{ij}|,\ i \ne j$. The matrix A is said to be an H–matrix if $\langle A\rangle$ is a nonsingular M–matrix. The splitting $A = M - N$ is called a regular splitting if $M^{-1} \ge O$ and $N \ge O$, it is called a weak regular splitting if $M^{-1} \ge O$ and $M^{-1}N \ge O$; the splitting is an H–splitting if $\langle M\rangle - |N|$ is an M–matrix, the splitting is an H–compatible splitting if $\langle A\rangle = \langle M\rangle - |N|$; see e.g., [4]. Let x be a positive vector, we consider the vector norm $\|y\|_x = \inf\{\beta > 0 \ :\ |y| \le \beta x\}$. This vector norm is monotone increasing and for every matrix $B \in \mathbb{R}^{n\times n}$ it satisfies $\|\ |B|x\|_x = \|B\|_x$, where $\|B\|_x$ denotes the matrix norm of B induced by the vector norm above defined.

Lemma 1 [6] *Let $A = M - N$ be an H–splitting. Then A and M are H–matrices and $\rho(M^{-1}N) \leq \rho(\langle M\rangle^{-1}|N|) < 1$.*

Lemma 2 [6] *Let $A = M - N$ be an H–compatible splitting and assume that A is an H–matrix. Then $A = M - N$ is an H–splitting and thus the splitting is convergent.*

Theorem 1 [11] *Let A be an $n \times n$ M–matrix, then for every zero pattern subset S of $S_n = \{(i,j) : i \neq j,\ 1 \leq i,j \leq n\}$, there exist a unit lower triangular matrix $L = (l_{ij})$, an upper triangular matrix $U = (u_{ij})$, and a matrix $N = (n_{ij})$ with $l_{ij} = u_{ij} = 0$ if $(i,j) \in S$ and $n_{ij} = 0$ if $(i,j) \notin S$, such that $A = LU - N$ is a regular splitting of A. Moreover, the factors L and U are unique.*

Theorem 2 [9] *Let A be an $n \times n$ H–matrix. Let $A = LU - N$ and $\langle A\rangle = \hat{L}\hat{U} - \hat{N}$ be the ILU factorizations of A and $\langle A\rangle$ corresponding to a zero pattern subset S of $S_n = \{(i,j) :\ i \neq j,\ 1 \leq i,j \leq n\}$, respectively. Then $|L^{-1}| \leq \hat{L}^{-1},\ |U^{-1}| \leq \hat{U}^{-1},\ |(LU)^{-1}N| \leq (\hat{L}\hat{U})^{-1}\hat{N}$.*

2. Convergence

In order to obtain the two–stage multisplitting of the Jacobian matrix $F'(x)$, we consider the following outer splittings $F'(x) = P_k(x) - Q_k(x),\ 1 \leq k \leq p$. Then we perform ILU factorizations of the matrices $P_k(x)$. This entails for each $k,\ 1 \leq k \leq p$, a decomposition of the form $P_k(x) = M_k(x) - N_k(x)$, where $M_k(x) = L_k(x)U_k(x)$, and the matrices $L_k(x)$ and $U_k(x)$ are unit lower triangular and upper triangular matrices, respectively, and $N_k(x)$ is the residual or error of the factorization. This incomplete factorization is rather easy to compute. A general algorithm for building ILU factorizations can be derived by performing Gaussian elimination and dropping some elements in predetermined nondiagonal positions (see e.g., [11]). As was done in [1], to study the convergence of the iterative scheme (6) setting $M_k(x) = L_k(x)U_k(x)$, we need to make the following assumptions:

(i) There exists an $r_0 > 0$ such that F is differentiable on $\mathcal{S}_0 \equiv \{x \in \mathbb{R}^n\ :\ \|x - x^*\| < r_0\}$,

(ii) the Jacobian matrix at x^*, $F'(x^*)$, is nonsingular,

(iii) there exists a $\vartheta > 0$ such that for $x \in \mathcal{S}_0,\ \|F'(x) - F'(x^*)\| \leq \vartheta\|x - x^*\|$,

(iv) $P_k(x)$ and $M_k(x),\ 1 \leq k \leq p$, are Lipschitz–continuous at x^*, i.e., there exist positive constants $\mu_k,\ \eta_k,\ 1 \leq k \leq p$, such that, for $x \in \mathcal{S}_0$, $\|P_k(x) - P_k(x^*)\| \leq \mu_k\|x - x^*\|$, $\|M_k(x) - M_k(x^*)\| \leq \eta_k\|x - x^*\|$,

(v) $P_k(x^*)$ and $M_k(x^*),\ 1 \leq k \leq p$, are nonsingular,

(vi) there exists $0 \leq \alpha < 1$, such that, for each positive integer s and $\ell = 0, 1, \ldots$, $\|H_{\ell,s}(x^*)\| \leq \alpha$, where $H_{\ell,s}(x)$ is defined in (5).

Theorem 3 [1] *Let assumptions (i)–(vi) hold and $F(x^*) = 0$. Let $\{m_\ell\}_{\ell=0}^{\infty}$ be a sequence of positive integers, and define*

$$m = \max\left[\{m_0\} \cup \left\{m_\ell - \sum_{i=0}^{\ell-1} m_i\ :\ \ell = 1, 2, \ldots\right\}\right]. \tag{8}$$

Suppose that $m < +\infty$ and that the sequence of number of inner linear iterations $q(\ell, s, k),\ \ell = 0, 1, \ldots,\ s = 1, 2, \ldots, m_\ell,\ 1 \leq k \leq p$, is bounded by $q > 0$. Then, there exist $r > 0$ and $c < 1$ such that, for $x^{(0)} \in \mathcal{S} \equiv \{x \in \mathbb{R}^n\ :\ \|x - x^*\| < r\}$, the sequence of iterates defined by (6) converges to x^* and satisfies $\|x^{(\ell+1)} - x^*\| \leq c^{m_\ell}\|x^{(\ell)} - x^*\|$.

We note that if the inner splittings satisfy $\|M_k(x^*)^{-1}N_k(x^*)\| < 1,\ 1 \leq k \leq p$, Theorem 3 remains valid without the assumption that the sequence of number of inner linear iterations $q(\ell, s, k)$, $\ell = 0, 1, \ldots,\ s = 1, 2, \ldots, m_\ell,\ 1 \leq k \leq p$, be bounded (see Remark 4.3 of [1]).

Theorem 4 *Let assumptions (i)–(iv) hold and $F(x^*) = 0$. Let $\{m_\ell\}_{\ell=0}^{\infty}$, be a sequence of positive integers, and define m as in (8). Suppose that $m < +\infty$. Let $F'(x^*)$ be an H–matrix, and the splittings $F'(x^*) = P_k(x^*) - Q_k(x^*),\ 1 \leq k \leq p$, be H–compatible. Let $P_k(x^*) = M_k(x^*) - N_k(x^*),\ 1 \leq k \leq p$, where $M_k(x^*) = L_k(x^*)U_k(x^*)$ is the ILU factorization of $P_k(x^*)$ corresponding to a zero pattern subset S_k of $S_n = \{(i, j) :\ i \neq j,\ 1 \leq i, j \leq n\}$. Then, there exist $r > 0$ and $c < 1$ such that, for $x^{(0)} \in \mathcal{S} \equiv \{x \in \mathbb{R}^n\ :\ \|x - x^*\| < r\}$, the sequence of iterates defined by (6) converges to x^* and satisfies $\|x^{(\ell+1)} - x^*\| \leq c^{m_\ell}\|x^{(\ell)} - x^*\|$.*

Proof: From Theorem 3 it follows that it suffices to prove that there exists $\alpha < 1$ such that $\|(L_k(x^*)U_k(x^*))^{-1}N_k(x^*)\| \leq \alpha,\ 1 \leq k \leq p$, and $\|H_{\ell,s}(x^*)\| \leq \alpha,\ \ell = 0, 1, 2, \ldots,\ s = 1, 2, \ldots, m_\ell$, for some matrix norm.

Since $F'(x^*) = P_k(x^*) - Q_k(x^*),\ 1 \leq k \leq p$, are H–compatible splittings of an H–matrix, from Lemmata 1 and 2, it follows that $P_k(x^*),\ 1 \leq k \leq p$, are H–matrices. Then $\langle P_k(x^*)\rangle,\ 1 \leq k \leq p$, are M–matrices. Following Theorem 1 there exists an ILU factorization of $\langle P_k(x^*)\rangle$ corresponding to the zero pattern subset S_k such that $\langle P_k(x^*)\rangle = \hat{L}_k(x^*)\hat{U}_k(x^*) - \hat{N}_k(x^*),\ 1 \leq k \leq p$, are regular splittings.

Let us denote $\hat{R}_k(x) = (\hat{L}_k(x)\hat{U}_k(x))^{-1}\hat{N}_k(x)$, then taking into account (4) y (5), from Theorem 2 it follows

$$|H_{\ell,s}(x^*)| \leq \sum_{k=1}^{p} E_k \hat{R}_k^{q(\ell,s,k)}(x^*) + \sum_{k=1}^{p} E_k \left(\sum_{i=0}^{q(\ell,s,k)-1} \hat{R}_k^i(x^*) \right) (\hat{L}_k(x^*)\hat{U}_k(x^*))^{-1}|Q_k(x^*)| \equiv \hat{H}_{\ell,s}(x^*).$$

That is, $|H_{\ell,s}(x^*)| \leq \hat{H}_{\ell,s}(x^*)$, where $\hat{H}_{\ell,s}(x^*)$ are the iteration matrices of a two–stage multisplitting method for the splittings $\langle F'(x^*)\rangle = \langle P_k(x^*)\rangle - |Q_k(x^*)|,\ 1 \leq k \leq p$ and $\langle P_k(x^*)\rangle = \hat{L}_k(x^*)\hat{U}_k(x^*) - \hat{N}_k(x^*),\ 1 \leq k \leq p$. Moreover these splittings are regular.

We consider the vector $e = (1, \ldots, 1)^T$, since $\langle F'(x^*)\rangle^{-1} \geq O$, it follows that $u = \langle F'(x^*)\rangle^{-1}e > 0$, and we can assure (see e.g., [5]) that there exists $0 \leq \alpha < 1$ such that $|H_{\ell,s}(x^*)|u \leq \hat{H}_{\ell,s}(x^*)u \leq \alpha u$. Hence $\|H_{\ell,s}(x^*)\|_u \leq \alpha,\ \ell = 0, 1, \ldots,\ s = 1, \ldots, m_\ell$. On the other hand, setting the fixed vector $u = \langle F'(x^*)\rangle^{-1}e$, it obtains $(\hat{L}_k(x^*)\hat{U}_k(x^*))^{-1}\hat{N}_k(x^*)u \leq u - (\hat{L}_k(x^*)\hat{U}_k(x^*))^{-1}e < u$.

Then, from Theorem 2 it follows that $|(L_k(x^*)U_k(x^*))^{-1}N_k(x^*)|u < u$, and therefore for $1 \leq k \leq p$, $\|(L_k(x^*)U_k(x^*))^{-1}N_k(x^*)\|_u < 1$. Thus the proof is complete.

In the parallel Newton two–stage method (6), when ILU factorizations are used in order to construct the two–stage multisplitting, the Jacobian matrix must be computed at the current iterate, and a portion of this matrix ($P_k(x^{(\ell)})$) must be factored at each non–linear iteration. One approach to reduce the cost of each non–linear iteration is to consider the sequence of iterates $x^{(\ell+1)} = \hat{G}_{\ell,m_\ell}(x^{(\ell)}, x^{(0)})$, where $\hat{G}_{\ell,m_\ell}(x, y) = x - A_{\ell,m_\ell}(y)F(x)$ and $A_{\ell,m_\ell}(y)$ is defined in (7).

This method, based on the Chord method (see e.g., [10]), will be called the parallel Chord two–stage method. The only difference in implementation from the parallel Newton two–stage method is that the computation and, therefore, the obtaining of the two–stage multisplitting of the Jacobian matrix is done before the nonlinear iteration begins. This technique could be interesting to reduce the computational time. The difference in the iteration itself is that another approximation to $F'(x^{(\ell)})$ is used. Another technique consists of alternating a Newton step with a sequence of Chord steps. In this case we can describe the transition from $x^{(\ell)}$ to $x^{(\ell+1)}$ by

For $\ell = 0, 1, 2, \ldots$
If mod$(\ell, \tau) = 0$ then $y = x^{(\ell)}$
$x^{(\ell+1)} = \hat{G}_{\ell,m_\ell}(x^{(\ell)}, y)$.

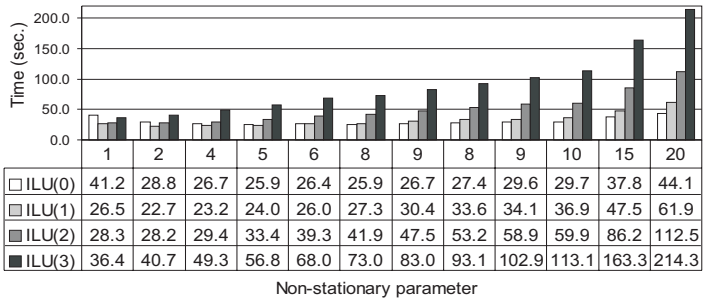
This method, based on the Shamanskii method (see e.g., [10]), will be called the parallel Shamanskii two–stage method. Note that $\tau = 1$ is the parallel Newton two–stage method and $\tau = \infty$ is the corresponding Chord method.

It is not difficult to see that the convergence results of Theorem 3, and therefore, those of Theorem 4, remain valid for both variations of the Newton method described above.

3. Numerical experiments

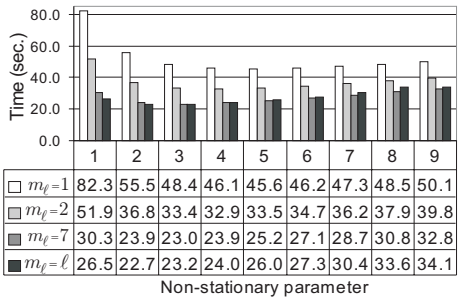
In order to illustrate the behaviour of these methods, we implemented the algorithms described here on two distributed memory multiprocessors. The first platform is an IBM RS/6000 SP with 8 nodes. These nodes are 120 MHz Power2 Super Chip and are connected through a high performance switch with latency time of 40 microseconds and a bandwidth of 110 Mbytes per second. The second platform is an Ethernet network of 6 Pentiums IV connected through a switch with a bandwidth of 1 Gbit per second (Cluster 1 Gb/sec.). The parallel environment has been managed using the MPI library of parallel routines. Moreover, we have used the BLAS routines for vector computations and the SPARSKIT routines for handling sparse matrices. In order to illustrate the behaviour of the above algorithms, we have considered a nonlinear elliptic partial differential equation, known as the Bratu problem. In this problem, heat generation from a combustion process is balanced by heat transfer due to conduction. The model problem is given as $\nabla^2 u - \lambda e^u = 0$, where u is the temperature and λ is a constant known as the Frank–Kamenetskii parameter (see e.g., [3]). There are two possible steady–state solutions to this problem for a given value of λ. One solution is close to $u = 0$ and it is easy to obtain. A close starting point is needed to converge to the other solution. For our model case, we consider a 3D square domain Ω of unit length and $\lambda = 6$. To solve our model problem using the finite difference method, we consider a grid in Ω of d^3 nodes. This discretization yields a nonlinear system of the form $F(x) = Ax + \Phi(x) - b = 0$, where $\Phi : \mathbb{R}^n \to \mathbb{R}^n$ is a nonlinear diagonal mapping (i.e., the ith component Φ_i of Φ is a function only of x_i). The Jacobian matrix is a sparse matrix of order d^3 and the typical number of nonzero elements per row of this matrix is seven, with fewer in rows corresponding to boundary points of the physical domain. In our experiments, we have considered the outer splittings $F'(x) = P_k(x) - Q_k(x)$ determined by $P_k(x) = \text{diag}(I, \ldots, D_k(x), \ldots, I),\ 1 \le k \le p$, where $D_k(x)$ consists of the k diagonal block of $F'(x)$ of size n_k, and the inner splittings $P_k(x) = M_k(x) - N_k(x)$ are determined by $M_k(x) = \text{diag}(I, \ldots, L_k(x)U_k(x), \ldots, I),\ 1 \le k \le p$, where $L_k(x)U_k(x)$ is the "level of fill–in" factorizations ILU of $D_k(x)$ [11]. Each diagonal weighting matrix E_k has ones in the entries corresponding to the diagonal block $D_k(x)$ and zero otherwise. Note that, with this choice of the two-stage multisplitting, each processor k only needs to approximate, at each inner iteration, linear systems of size n_k, where $\sum_{k=1}^{p} n_k = n$, and n is the size of the problem to be solved. In order to preserve the block structure of the Jacobian matrices we have considered, in our experiments, n_k a multiple of d.

Let us denote by ILU(S) the incomplete LU factorization associated with the zero pattern subset S of $S_n = \{(i, j) :\ i \neq j,\ 1 \le i, j \le n\}$. In particular, when $S = \{(i, j) :\ a_{ij} = 0\}$, the incomplete factorization with null fill–in is known as ILU(0). To improve the quality of the factorization, many strategies for altering the pattern have been proposed. In the "level of fill–in" factorizations [8], ILU(κ), a level of fill–in is recursively attributed to each fill–in position from the levels of its parents. Then, the positions of level lower than κ are removed from S. In the experiments reported here, we



	1	2	4	5	6	8	9	8	9	10	15	20
ILU(0)	41.2	28.8	26.7	25.9	26.4	25.9	26.7	27.4	29.6	29.7	37.8	44.1
ILU(1)	26.5	22.7	23.2	24.0	26.0	27.3	30.4	33.6	34.1	36.9	47.5	61.9
ILU(2)	28.3	28.2	29.4	33.4	39.3	41.9	47.5	53.2	58.9	59.9	86.2	112.5
ILU(3)	36.4	40.7	49.3	56.8	68.0	73.0	83.0	93.1	102.9	113.1	163.3	214.3

(a) $m_\ell = \ell$.



	1	2	3	4	5	6	7	8	9
m_ℓ=1	82.3	55.5	48.4	46.1	45.6	46.2	47.3	48.5	50.1
m_ℓ=2	51.9	36.8	33.4	32.9	33.5	34.7	36.2	37.9	39.8
m_ℓ=7	30.3	23.9	23.0	23.9	25.2	27.1	28.7	30.8	32.8
$m_\ell = \ell$	26.5	22.7	23.2	24.0	26.0	27.3	30.4	33.6	34.1

(b) $\kappa = 1$.

Figure 1. Parallel Newton two–stage method, $p = 4,\ n = 125000$, Cluster 1Gb/sec.

have used these ILU(κ) factorizations for the matrices $D_k(x),\ 1 \leq k \leq p$, defined above. We have modified the SPARSKIT routine which obtains the ILU(κ) factorization in order to improve the factorizations of successive matrices with the same sparsity pattern. The stopping criterion used was $\|F(x^{(\ell)})\|_2 < 10^{-\delta}$, with $\delta = 7$. All times are reported in seconds.

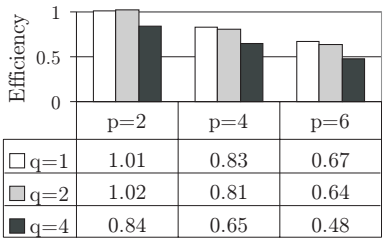
We have run our codes with problems of various sizes and different levels of fill–in for the ILU factorizations. In order to focus our discussion, we present here results obtained with $d = 50$ and $d = 72$ that lead to nonlinear systems of size $n = 125000$ and $n = 373248$, respectively.

Figure 1(a) illustrates, using four processors, the influence on the execution time of different levels of fill–in for $m_\ell = \ell$. It can be observed that the best times are obtained setting incomplete LU factorizations with levels $\kappa = 0$ or $\kappa = 1$, depending on the number of local steps $q(\ell, s, k) = q$ performed. That is, for small values of q (approximately $q \leq 5$, in this figure) the best times were obtained setting $\kappa = 1$, while for high values of q it should use level 0 of fill–in. The levels of fill–in ILU(κ) refer to the amount of fill–in allowed during the incomplete factorization. For given values of m_ℓ and q, increasing the level of fill–in κ, provides a better quality method in terms of its rate of convergence (that is, in terms of reducing the number of global iterations required), but at the cost of increasing storage. Therefore, it can be seen that, for $\kappa > 1$ this reduction of global iterations does not balance the increase of the computational cost obtained by increasing the level of fill–in.

Figure 1(b) illustrates the influence of the choice of $m_\ell,\ \ell = 0, 1, \ldots,$ in relation to the non–stationary parameter q used, for $\kappa = 1$. As it can be appreciated, the optimal choice of the values $m_\ell,\ \ell = 0, 1, \ldots$ and q, are $m_\ell = \ell$ and small values of q, respectively. However, if we use values of q higher than the optimal, the method behaves better using a constant value of linear iterations m_ℓ at each global iteration ℓ, but this optimal value is hard to predict.

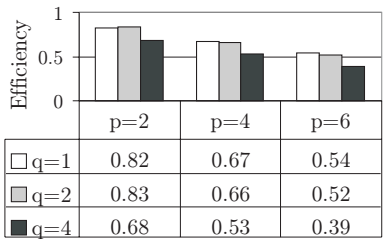
We have compared the results of these parallel methods with the well–known sequential Newton Gauss–Seidel method, the sequential Newton ILU method [10], and with respect to the same algorithm executed on a single processors. In all cases, the best sequential methods were obtained with $m_\ell = \ell$. Figure 2 illustrates the obtained efficiencies. As it can be appreciated, the parallel implementations reduce substantially the sequential times. Also, these efficiencies show a good degree of parallelism of the method treated here.

Nevertheless these algorithms need to perform the evaluation and factorization of the Jacobian at each nonlinear iteration; this step is one of the most costly. As it can be seen in Section 2, one



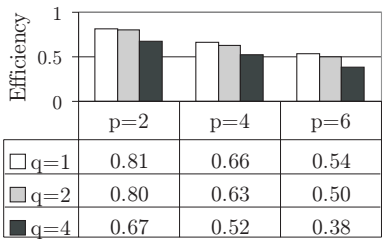
	p=2	p=4	p=6
q=1	1.01	0.83	0.67
q=2	1.02	0.81	0.64
q=4	0.84	0.65	0.48

(a) Sequential reference algorithm: Newton Gauss-Seidel.



	p=2	p=4	p=6
q=1	0.82	0.67	0.54
q=2	0.83	0.66	0.52
q=4	0.68	0.53	0.39

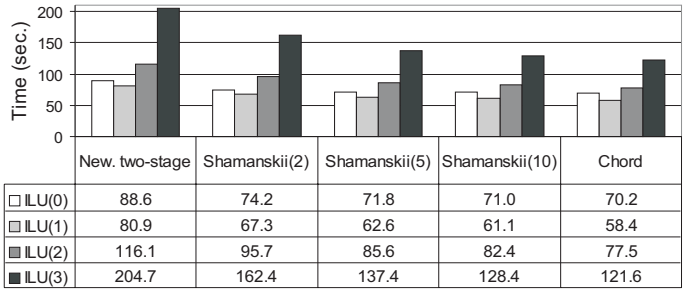
(b) Sequential reference algorithm: Newton ILU.



	p=2	p=4	p=6
q=1	0.81	0.66	0.54
q=2	0.80	0.63	0.50
q=4	0.67	0.52	0.38

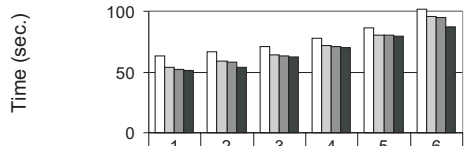
(c) Sequential reference algorithm: Newton two–stage, $p = 1$.

Figure 2. Efficiency of parallel Newton two–stage method, $m_\ell = \ell,\ \kappa = 1,\ n = 125000$, IBM RS/6000 SP.



	New. two-stage	Shamanskii(2)	Shamanskii(5)	Shamanskii(10)	Chord
ILU(0)	88.6	74.2	71.8	71.0	70.2
ILU(1)	80.9	67.3	62.6	61.1	58.4
ILU(2)	116.1	95.7	85.6	82.4	77.5
ILU(3)	204.7	162.4	137.4	128.4	121.6

(a) $m_\ell = 7,\ q = 3$.



	1	2	3	4	5	6
New. two-stage	63.2	66.8	70.7	77.4	86.5	101.3
Shamanskii(5)	53.6	59.4	64.3	71.6	80.7	95.4
Shamanskii(10)	52.4	58.5	63.4	70.7	80.1	94.8
Chord	51.5	54.0	62.7	70.0	79.5	86.8

Non-stationary parameter

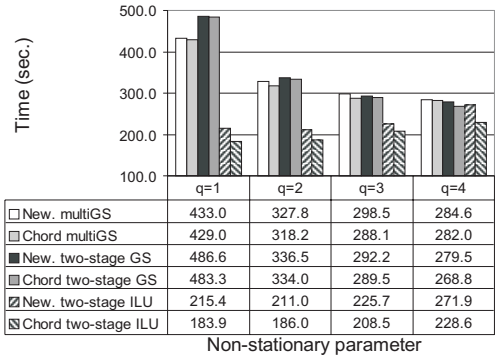
(b) $m_\ell = \ell,\ \kappa = 1$.

Figure 3. Parallel variations of Newton two–stage method, $p = 3,\ n = 125000$, IBM RS/6000 SP.

approach to reduce the cost of each nonlinear iteration of a Newton algorithm can be obtained with far fewer Jacobian evaluations or factorizations (Shamanskii method), or with only one Jacobian evaluation, before the nonlinear iteration begins (Chord method).

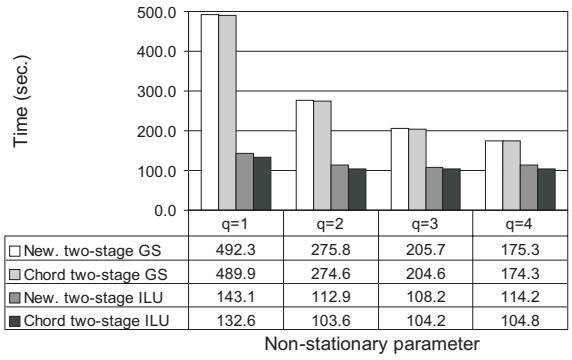
Figure 3 illustrates the behaviour of these variations of the parallel Newton method studied here. In this figure, Shamanskii(τ) indicates that the Jacobian matrix is updated each τ nonlinear iterations. The results obtained indicate that the parallel Chord methods are the best option for this problem. Moreover, the best results were obtained, again, with level 1 of fill–in (i.e., using ILU(1)).

On the other hand, we have compared the methods introduced here with both, those corresponding to the algorithms presented in [2] (denoted here by New. multiGS) and those corresponding to the algorithms presented in [1] (denoted here by New. two–stage GS). The splittings used in each case are the same as the ones used in [2] and [1], respectively. Moreover, we have implemented the corresponding Chord variation of those methods (denoted here Chord multiGS and Chord two–stage GS, respectively). The best results obtained for the methods introduced in [2] and [1], and its Chord variation, described here, have been obtained setting $m_\ell = \ell$. As it can be appreciated in Figure 4,



	q=1	q=2	q=3	q=4
New. multiGS	433.0	327.8	298.5	284.6
Chord multiGS	429.0	318.2	288.1	282.0
New. two-stage GS	486.6	336.5	292.2	279.5
Chord two-stage GS	483.3	334.0	289.5	268.8
New. two-stage ILU	215.4	211.0	225.7	271.9
Chord two-stage ILU	183.9	186.0	208.5	228.6

(a) IBM RS/6000 SP.



	q=1	q=2	q=3	q=4
New. two-stage GS	492.3	275.8	205.7	175.3
Chord two-stage GS	489.9	274.6	204.6	174.3
New. two-stage ILU	143.1	112.9	108.2	114.2
Chord two-stage ILU	132.6	103.6	104.2	104.8

(b) Cluster 1Gb/sec.

Figure 4. Comparison of parallel Newton two–stage methods, $m_\ell = \ell,\ p = 6,\ \kappa = 1,\ n = 373248$.

the numerical experiments performed with these methods indicate that the use of parallel methods using ILU factorizations to obtain the splittings is preferable to the use of parallel Newton methods using the multisplittings described in [2] and [1] (Gauss-Seidel type splittings). Concretely, we have observed a substantial reduction in the computational time when solving our model problem. By comparing the best execution times in each figure, the parallel implementations of the methods introduced in this paper provide a time reduction of about 30%-60%.

References

[1] J. Arnal, V. Migallón, and J. Penadés. Parallel Newton two–stage multisplitting iterative methods for nonlinear systems. *BIT Numerical Mathematics*, 43:849–861, 2003.

[2] J. Arnal, V. Migallón, and J. Penadés. Non–stationary parallel Newton iterative methods for nonlinear problems. *Lecture Notes in Computer Science*, 1981:380–394, 2001.

[3] B.M. Averick, R.G. Carter, J.J. More, and G. Xue. The MINPACK–2 Test Problem Collection. TR MCS–P153–0692, Mathematics and Computer Science Division. Argonne National Laboratory, 1992.

[4] A. Berman and R.J. Plemmons. *Nonnegative Matrices in the Mathematical Sciences*. Academic Press, New York, third edition, 1979. Reprinted by SIAM, Philadelphia, 1994.

[5] R. Bru, V. Migallón, J. Penadés, and D.B. Szyld. Parallel, synchronous and asynchronous two–stage multisplitting methods. *Electronic Transactions on Numerical Analysis*, 3:24–38, 1995.

[6] A. Frommer and D.B. Szyld. H–splittings and two–stage iterative methods. *Numerische Mathematik*, 63:345–356, 1992.

[7] D.A. Knoll and D.E. Keyes. Jacobian-free Newton–Krylov methods: a survey of approaches and applications. *Journal of Computational Physics*, 193:357–397, 2004.

[8] H.P. Langtangen. Conjugate gradient methods and ILU preconditioning of nonsymmetric matrix systems with arbitrary sparsity patterns. *International Journal for Numerical Methods in Fluids*, 9:213–233, 1989.

[9] A. Messaoudi. On the stability of the incomplete LU factorizations and Characterizations of H–matrices. *Numerische Mathematik*, 69:321–331, 1995.

[10] J.M. Ortega and W.C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, San Diego, 1970.

[11] Y. Saad. *Iterative Methods for Sparse Linear Systems*. PWS Publishing Company, 1996.